IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: Kevin Marshall

Application No.: 10/052,784

Filed: November 2, 2001

Title: METHODS AND APPARATUS FOR
DETERMINING SOFTWARE COMPONENT
SIZES ASSOCIATED WITH ERRORS

Attorney Docket No.: SUN1P749

Examiner: KENDALL, Chuck O.

Group: 2192

Confirmation No.: 3211

PRE-APPEAL BRIEF REQUEST FOR REVIEW

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Applicant hereby requests review of the rejections in the above-identified application. This request is being filed with a Notice of Appeal. Review is requested for the reasons stated in the accompanying remarks.

Applicant filed a response after the final rejection. Applicant's response was filed before the two month period (on December 20, 2005), but Applicant has not received an Advisory Action. Applicant's undersigned attorney (through his administrative assistant) left several messages for the Examiner during the past few weeks, but no response was received. This review is thus based on the substance of the final rejection, without benefit of an Advisory Action.

REMARKS

For the purpose of this PRE-APPEAL BRIEF REQUEST FOR REVIEW only, Applicant treats claim 1 as a representative claim. In specific instances below, Applicant also addresses features recited in other claims, including claims dependent on claim 1. Applicant reserves the right to argue each claim separately when and if an Appeal Brief is eventually filed.

Summary of Subject Matter of Claims 1 and 2

As discussed in the Applicant's specification, in the Summary (page 5):

Contents of one or more files indicating errors in the software system are examined to determine software components responsible for the errors. In addition, the number of errors attributed to each of the software components

1

responsible for the errors is determined. A size of the software components responsible for one or more errors is also determined.

In one embodiment, the size of the software components responsible for errors is correlated with the number of errors attributed to the software components. This automatically generated information provides the data which is useful to assess the quality of the software code and the probability of errors in code software components. This data is also useful as a basis for subsequent reliability studies.

Summary of the Obviousness Rejections

Referring to page 3 of the Office Action dated October 20, 2005 (Final Office Action), the Examiner essentially contends that Chung discloses all of the elements of claim 1, except that "Chung does not expressly disclose said errors being generated during execution of the software system." The Examiner relies on Smith for its alleged disclosure of "Smith et al. disclose a method of identifying errors generated during execution ..." The Examiner contends that one would be motivated to modify Chung in view of Smith "to ensure the library program functions associated with the program can be called dynamically."

Clear Error in the Examiner's Rejection of Claims

Even taking as correct (for the purpose of argument only) the Examiner contention that the combination of Chung and Smith yields what is claimed, it is respectfully submitted that the Examiner has not stated a proper motivation to combine these references. It is further respectfully submitted that, in any event, there *is* no proper motivation to combine these references.

In the first place, while the Examiner contends that both Chung and Smith are "directed to monitoring, identifying errors in software programs," this is not, in fact, the case. Rather, Chung is directed to a method to assist in time and labor planning for debugging and testing a program, not <u>actually</u> debugging and testing the program. As discussed in Applicant's previous response, much of the latter part of the Chung disclosure is dedicated to describing a formula for applying the number of errors found in a subset of code to estimate the total amount of time and labor required to test and debug the entire code (taking into consideration, for example, the skill level of the developer).

To accomplish the estimate, Chung discloses "scanning" (as the Examiner recognizes, not "executing") a sample portion of the program for errors and, based on the found errors and the total size of the program, estimating the total number of errors in the program. Contrary to the Examiner's assertion (with respect to asserting that Chung and Smith are analogous art), this is not monitoring errors at all. And, to the extent it is "identifying" errors, the errors (more specifically, errors in a sample portion of the program) are identified only for the purpose of estimating the total number of errors.

Again, the Chung disclosure is directed only to estimating the total number of errors, based on the errors in a small portion of the code, for estimating the total amount of time and

labor required to test and debug the code. The real work of identifying and fixing all the errors begins after where the Chung disclosure leaves off.

With respect to Smith, Applicant agrees that Smith discloses testing and debugging programs for identifying errors generated during execution. With respect to this broad aspect, Smith discloses nothing more than programmers have been doing since the beginning of programming time.

As discussed above, given that Chung is directed to estimating, for time and labor planning (with actual test and debugging to follow), one would not be motivated to modify Chung to include the "actual testing" features disclosed by Smith. Such "actual testing" is not necessary, or even desired, for Chung to calculate an estimate for time and labor planning. In fact, including the "actual testing" is counter to Chung's intended purpose, since Chung is directed to estimating the time and labor required to test and debug the code *without* actually testing the code.

More particularly, if the time and labor required to test and debug the code were determined based on actual testing, then there would be no need to estimate the time and labor required to test and debug the code, since such testing and debugging would have already been performed, and actual measurements of the time and labor to test and debug the code would be readily available.

Curiously, the Examiner states that the alleged motivation for modifying Chung in view of Smith would be "to ensure the library program functions associated with the program can be called dynamically." But, Chung is not concerned with ensuring anything, only with estimating time and labor for later debugging and testing a program. If the Examiner continues to reject the claims based on Chung and Smith, Applicant would appreciate the Examiner more specifically explaining his contention that one would be motivated to modify the Chung disclosure "to ensure that library program functions associated with the program can be called dynamically." This alleged motivation appears to be completely unfounded from the references or from knowledge held by one of ordinary skill in the art and, therefore, can only be completely based on hindsight reasoning.

With regard to the other references relied upon, the Examiner further relies on Ruhlen in rejecting some of the claims. Contrary to the Examiner's assertion, Applicant can find nothing in Ruhlen that discloses storing modifications made in response to errors. Fig. 2 and corresponding text, cited by the Examiner, discloses compiling and storing information of failures, but there is no disclosure of storing modifications made in response to errors. Neither does the disclosure at col. 3, lines 63-37 or col. 2, lines 11-13 describe storing modifications made in response to errors.

In response to the first Office Action, Applicant pointed out that Ruhlen does not disclose storing modifications made in response to errors, specifically discussing the portions of Ruhlen

cited by the Examiner. In the Final Office Action, the Examiner has merely repeated the statements regarding Ruhlen.

Similarly, with regard to Leung, Applicant pointed out in response to the first Office Action that Leung discloses a testing tool inserted into the code to be tested, so that it can be determined where errors are occurring. Applicant also pointed out that, given that this has nothing to do with Chung's disclosure of time and labor planning, it is respectfully submitted that there is no proper motivation to modify Chung in view of Leung. Applicant also argued that, similarly, there is no proper motivation to modify Chung in view of Hanson.

CONCLUSION

It is respectfully submitted that Examiner's rejections are in clear error and that this application is in condition for allowance. Notice to that effect is earnestly solicited.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP

/ASH/
Alan S. Hodes
Reg. No. 38,185

P.O. Box 70250
Oakland, CA 94612-0250
(650) 961-8300